Amendments to the Drawings:

Please replace Figures 1 and 2 with the attached replacement sheets.

Attachment: Replacement Sheet 2 page(s)

5

REMARKS

Amendments to the Drawings

A typographical error is corrected in Figures 1 and 2. No new matter is added.

Claims 1-3, 5-11, 13, 15-17, 19-22 are rejected under 35 U.S.C. 103(a) as being unpatentable over Kaiser et al. (US Patent #5,784,710) in view of Hasbun (US Patent #6,205,458). Claims 4, 12, 14 and 18 are rejected under 35 U.S.C. 103(a) as being unpatentable over the combination of Kaiser et al. (US Patent #5,784,710) and Hasbun (US Patent #6,205,458) references as applied to claims 2, 5, 9 and 11 above, and further in view of DeBruler (US Patent #4,539,637).

Claims 1-22 are cancelled without disclaimer to the subject matter thereof.

New Claims

New claims 23 – 42 are added. Claims 23 and 28 are independent method claims, and claims 33 and 38 are independent system claims. No new matter is added. In the following sections, applicant has provided support for each new claim in the specification as originally filed and comments regarding the patentability with respect to the cited references of Hasbun and Kaiser et al.

Firstly, concerning independent claims 23 and 33, applicant points out that Figure 2 as originally filed includes a memory 60, a section of memory 62 to be protected, a microprocessor 40, and an address translator for shifting the first memory address 42 by

an amount being substantially equal to the size of the memory section 62 to thereby generate a shifted memory address (output of operating unit 54) being different from the first memory address 42. Supporting the shifting limitation of new claim 23, paragraph [0037] states, "This is because the first logic address data is shifted by the address translator 50 with an amount equal to the size of the boot code section 62." Additionally, paragraph [0034] of the specification states, "the operating unit 54 of the address translator 50 is an adder. The operating unit 54 adds the setup value (04000H) to the first logic address data (00000H), which results in the second logic address data (04000H)." Therefore, the resulting second logic address data is a shifted memory address. Applicant notes that when the address translator is switched on, only the shifted memory address is coupled to the memory and therefore all memory accesses are performed utilizing the shifted memory address. In this way, the boot section 62 is protected. Such operation is supported in paragraph [0037] stating, "Thus the microprocessor 40 cannot access the data in the boot code section 62 nor delete or update them. Therefore, the data in the boot code section 62 is safe from being accessed, deleted, or updated incorrectly when the address translator 50 is turned on."

Concerning the patentablity of new claims 23 and 33 with respect to the teachings of the cited references of Hasbun and Kaiser et al., applicant notes that neither cited reference teaches the following limitation:

> "accessing the memory utilizing the shifted memory address <u>for all memory accesses by the microprocessor</u> when it is desired to prevent the memory section from being accessed by the microprocessor" (claim 23 – emphasis added)

Concerning Kaiser et al., applicant points out that Kaiser et al. only teaches using the translated address when trying to access the IPL section. For example, see col 4, lines 29-36 stating, "If the address received from either processor 108 or processor 109 is an address intended to access the IPL code, but the processor sourcing the address is not

capable of addressing the IPL code, then circuitry 20 will effectively translate the received address from the processor into an address capable of accessing this IPL code. This translation is valid only if the target memory range is being accessed by the processor." (emphasis added) Applicant notes that the translation being valid only if the target memory range is being accessed by the processor is not equivalent or similar to the present invention as claimed in claim 23 stating, "accessing the memory utilizing the shifted memory address for all memory accesses by the microprocessor when it is desired to prevent the memory section from being accessed by the microprocessor". And note is again made that the shifted memory address is claimed to be different than the first address data generated by the microprocessor. For this reason, applicant does not expect the Examiner to interpret that Kaiser et al. is the actually equivalent to the above claimed operation due to the shifted address being the same as the first address data for some portions of memory.

Concerning the teachings of Hasbun, similar to the above comments regarding Kaiser et al., applicant points out that Hasbun only teaches swapping the memory addresses of two blocks of memory. For example, Hasbun only describes an embodiment where the address decoder 320 simply swaps block one 342 with block two 344 in the nonvolatile memory 340. In this way memory access by the microprocessor 330 for block 3 to block 5 will involve the address being outputted by the microprocessor 330 being the same as the address outputted by the address decoder utilized to access the memory 340. Such operation is described of in specific detail by Hasbun stating (col 5 line 64 to col 6 line 2), "... the swapping or remapping technique only affects a subset of the plurality of block in the nonvolatile memory so the address mapping to the other blocks remains unchanged. Thus for example, the block selector value has no effect on the mapping of addresses for block 346-350". Applicant notes that only swapping the addresses of two blocks in memory is not equivalent or similar to the present invention as claimed in claim 23 stating, "accessing the memory utilizing the shifted memory address for all memory accesses by the microprocessor when it is desired to prevent the memory section from

being accessed by the microprocessor". Again, applicant notes that the shifted memory address is claimed to be different than the first memory address so does not expect the Examiner to state that Hasbun is actually interpreted to be equivalent to the present invention because memory accesses to the memory can be made using the memory address outputted by the address translator 320 being the same as the memory address outputted by the microprocessor for some memory blocks 346-350 of the memory 340.

Because neither Kaiser et al. nor Hasbun teaches all the limitations of the present invention as claimed in claims 23 and 33, applicant asserts that a combination of the teachings of Kaiser et al. and Hasbun also cannot be deemed to teach all the limitations of the present invention without further inventive process. For at least this reason, applicant asserts that new claims 23 and 33 should be found allowable with respect to the teachings of Kaiser et al. and Hasbun. Consideration of claims 23 and 33 is respectfully requested.

Secondly, claims 24-27 and 34-37 are dependent upon claims 23 and 33, respectively, and are believed to be allowable for at least the same reasons as their base claims.

Concerning dependent claims 24, 29, 34 and 39, applicant remarks that the microprocessor is prevented from accessing the boot code (protected memory section) immediately after booting is complete, and for all subsequent normal operations. Paragraph [0046] states, "the microprocessor 40 executes a command predetermined in the last part of the boot code to have the controller 56 output the enable signal in order to turn on the address translator 50" and further states that it becomes "... not possible for the boot code stored in the boot code section 62 to be access nor erased or changed incorrectly by the microprocessor 40." No new matter is entered. Furthermore, claim 24 claims that the coupling of the shifted memory address to the memory after booting continues "... for all subsequent normal operations of the microprocessor." (emphasis added) This is neither taught by Kaiser et al nor Hasbun. In fact, Kaiser et al state in col 4, lines 21-24 that "if there is not a hit, circuitry outputs the originally received address to

the memory system" (emphasis added), and Hasbun teaches (in col 5, line 64 to col 6, line 2), "... the address mapping to the other blocks remains unchanged". (emphasis added) Similar arguments also apply to claims 29, 34 and 39. Consideration of claims 24, 29, 34 and 39 is respectively requested.

Concerning dependent claims 25, 30, 35 and 40, applicant notes that paragraph [0045] as originally filed describes the process of booting the microprocessor with the original memory address (that is, without shifting the memory address) before and during booting. Paragraph [0045] states, "... in order to have the microprocessor 40 execute the boot code stored in the boot code section 62 to boot the system, turn off the address translator 50 to have the boot code section 62 exist in the address space of the microprocessor 40." No new matter is entered. Consideration of claims 25, 30, 35 and 40 is respectively requested.

Concerning dependent claims 26, 31, 36 and 41, applicant describes a method by which the size of the memory section to be protected can be changed to a new size. Applicant refers to Paragraph [0038] which states that "... the address translator 50 can simply change the setup value stored in the register 52 to reduce the size of the protected memory section ..." (emphasis added). No new matter is entered. The new claim 26 is not taught by Hasbun, who specifically states (col 4, lines 60-65) that "the block sizes of the asymmetrically block flash, however, are determined when the nonvolatile memory is fabricated" (emphasis added), meaning they cannot be changed during operation. Hasbun continues (col 5, lines 28-30) stating, "the reference contents may be updated by swapping addresses with another unlocked block of the same size" (emphasis added) where in such cases, the teachings of Hasbun are still restricted to the block sizes of the memory which are determined at the time of manufacture. Consideration of claims 26, 31, 36 and 41 is respectively requested.

Concerning dependent claims 27, 32, 37 and 42, applicant has further presented the method by which boot code is stored in the memory section to be protected. Applicant points to Figure 2 as originally filed, which illustrates the boot code section 62 as the memory section to be protected. No new matter is entered. Consideration of claims 27, 32, 37 and 42 is respectively requested.

Concerning independent claims 28 and 38, applicant refers to Paragraph [0037] where applicant originally described that "the boot code section 62 can be regarded as not existing in the accessible address space of the microprocessor 40." No new matter is entered. Regarding the patentability of claims 28 and 38, neither Kaiser et al nor Hasbun disclose the following limitation:

> "accessing the memory utilizing shifted memory address being within the shifted memory address space for all memory accesses by the microprocessor when it is desired to prevent the memory section from being accessed by the microprocessor." (claim 28 – emphasis added)

Instead, the teachings of Kaiser et al state that, "This translation is valid only if the target memory range is being accessed by the processor." (col 4, lines 32-36) whereby only a subset of the memory range is translated, not the memory address space for all memory accesses. Regarding Hasbun, again please see col 5, lines 64-67, which teaches, "... the swapping or remapping technique only affects a subset of the plurality of block in the nonvolatile memory so the address mapping to the other blocks remains unchanged." Applicant asserts that translating a target memory range (as per Kaiser et al.) or swapping a subset (as taught by Hasbun) is not the same as moving an entire memory address space, as described in the present invention.

In addition, claims 28 and 38 state that, by way of the address translator 50 (when turned on with the control signal from controller 56), the entire memory address space

available to microprocessor 40 is shifted from the original memory address space to a shifted memory address space whereby the memory section 62 to be protected exists outside of this shifted memory address space. In this way, applicant asserts that the memory section (boot code section) 62 is rendered inaccessible by the microprocessor 40 and thus is protected from being accessed, deleted, or updated incorrectly (please see Paragraph [0037]). Neither Hasbun nor Kaiser et al teach shifting the memory address space "to thereby move the memory section to be protected outside of the shifted memory address space" (emphasis added).

Because neither Kaiser et al. nor Hasbun teaches all the limitations of the present invention as claimed in claims 28 and 38, applicant asserts that a combination of the teachings of Kaiser et al. and Hasbun also cannot be deemed to teach all the limitations of the present invention without further inventive process. For at least this reason, applicant asserts that new claims 28 and 38 should be found allowable with respect to the teachings of Kaiser et al. and Hasbun. Consideration of claims 28 and 38 is respectfully requested.

In addition, claims 29-32 and 39-42 are dependent upon claims 28 and 38, respectively, and are believed to be allowable for at least the same reasons as their base claims. For at least these reasons mentioned above, applicant asserts that newly added claims 23-42 should be found allowable with respect to the cited references. Consideration of new claims 23-42 is respectfully requested.

Conclusion:

Thus, all pending claims are submitted to be in condition for allowance with respect to the cited art for at least the reasons presented above. The Examiner is encouraged to telephone the undersigned if there are informalities that can be resolved in a phone conversation, or if the Examiner has any ideas or suggestions for further advancing the prosecution of this case.

Sincerely yours,

Winston Hsu Date: 09.17.2007

Winston Hsu, Patent Agent No. 41,526
P.O. BOX 506, Merrifield, VA 22116, U.S.A.
Voice Mail: 302-729-1562
Facsimile: 806-498-6673
e-mail : winstonhsu@naipo.com

Note: Please leave a message in my voice mail if you need to talk to me. (The time in D.C. is 12 hours behind the Taiwan time, i.e. 9 AM in D.C. = 9 PM in Taiwan.)